AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer implemented system for determining whether a packed executable is malware, the system comprising:

a malware evaluator for determining whether incoming data is malware, wherein the incoming data directed to a computing device is intercepted by the malware evaluator; and

an unpacking module that receives a first packed executable from the malware evaluator and returns ~~an~~ a corresponding substitute unpacked executable, without executing the first packed executable during unpacking, ~~corresponding to the entire packed executable;~~ the unpacking module comprising:

at least one substitute unpacker code segment, corresponding to a first unpacker code segment of the first packed executable, such that an appropriate substitute unpacker code segment is substituted for the first unpacker code segment of the received first packed executable to facilitate unpacking the first packed executable according to the substitute unpacker code rather than according to the first unpacker code, the first packed executable is thereby unpacked into a corresponding substitute unpacked executable that can be the same as, or different than, a first unpacked executable unpacked if the first packed executable were unpacked by the first unpacker code;

wherein the malware evaluator, upon receiving incoming data, can at least in part determine whether the incoming data is a packed executable, and if so, the malware evaluator provides the packed executable to the unpacking module such that ~~an~~ a substitute unpacked executable, corresponding to the first packed executable, is generated by unpacking the first packed executable with a substitute unpacker code segment without executing the first packed executable, whereby ~~can be received from the unpacking module, such that~~ the malware evaluator can determine whether the first unpacked executable ~~is~~ would be malware based at least in part on an analysis of the corresponding substitute unpacked executable.

2. (Withdrawn) A system for unpacking a packed executable for evaluation as malware, the system comprising:

a set of unpacker modules, the set of unpacker modules comprising at least one unpacker module and wherein each unpacker module corresponds to executable code for unpacking a particular type of packed executable; and

an unpacking manager, wherein the unpacking manager, upon obtaining a packed executable, selects an unpacker module from the set of unpacker modules to unpack the packed executable according to the type of the packed executable, and executes the selected unpacker module which generates an unpacked executable corresponding to the packed executable.

3. (Withdrawn) The system of Claim 2, wherein each unpacker module in the set of unpacker modules implements a confirmation interface routine for confirming whether the unpacker module is capable of unpacking the packed executable; and

wherein the unpacking manager selects an unpacker module from the set of unpacker modules to unpack the packed executable by:

iteratively calling the confirmation interface routine of each unpacker module in the set of unpacker modules until an unpacker module responds affirmatively to the call of its confirmation interface routine indicating that it can unpack the packed executable; and

selecting that unpacker module that responded affirmatively.

4. (Currently Amended) A method for determining whether incoming data is malware, the method comprising:

intercepting incoming data directed to a computing device;

determining whether the incoming data is a packed executable; and

if the incoming data is a packed executable:

accessing at least one substitute unpacker code segment corresponding to a first incoming packed executable of the incoming data;

substituting the substitute unpacker code segment for a first unpacker code segment of the first packed executable;

generating ~~an~~ a substitute unpacked executable employing the substitute unpacker code segment, the substitute unpacked executable corresponding to ~~the entire~~ a first unpacked ~~packed~~ executable that would result from unpacking the first packed executable with the first unpacker code; and

determining whether the first incoming packed executable is malware by evaluating whether the corresponding substitute unpacked executable ~~is~~ includes malware.

5. (Withdrawn) A method for unpacking a packed executable for evaluation as malware, the method comprising:

obtaining a packed executable;

selecting an unpacker module from a set of unpacker modules operable to unpack the packed executable; and

executing the selected unpacker module, thereby generating an unpacked executable corresponding to the packed executable.

6. (Withdrawn) An extensible unpacking module for unpacking a packed executable for evaluation as malware, the system comprising:

an set of unpacker modules comprising at least one unpacker module, wherein each unpacker module corresponds to executable code for unpacking a packed executable of a particular type, wherein the set of unpacker modules is dynamically extensible such that unpacker modules may be selectively added or removed to the set of unpacker modules; and

an unpacking manager, wherein the unpacking manager, upon obtaining a packed executable, selects an unpacker module from the set of unpacker modules to unpack the packed executable according to the type of the packed executable, and executes the selected unpacker module which generates an unpacked executable corresponding to the packed executable.

7. (Currently amended) The system of Claim 1, wherein the returned substitute unpacked executable corresponding to the packed first executable is based at least in part on code or data derived from employing ~~an~~ the substitute unpacker code rather ~~other~~ than the loader/unpacker received with the first packed executable.

8. (Currently amended) The system of Claim 7, wherein the employed substitute unpacker code is selected from a group of at least one modularized substitute unpacker modules germane to unpacking a first packed executable of a particular type and further germane to unpacking a first packed executable that has been intercepted by the malware evaluator.

9. (Previously Presented) The system of Claim 1, wherein the intercepted incoming data resides only in one or more logically or physically isolated memory stores such that the intercepted incoming data can be located at a computer but does not actually "reach" the computer.

10. (Previously Presented) The system of Claim 9, wherein the one or more isolated memory stores comprise at least one of a floppy disk, a flash memory storage device, magnetic tape, or combinations thereof.

11. (Currently amended) The system of Claim 1, wherein the corresponding substitute unpacked executable generated by the unpacking module corresponds to a complete first packed executable and not just a portion thereof.

12. (Currently amended) The system of Claim 11, wherein the corresponding generated substitute unpacked executable corresponding to a complete unpacked executable is unpacked without executing any portion thereof.

13. (Previously Presented) The system of Claim 1, wherein the malware evaluator determines whether the incoming data is malware without unpacking the incoming data if the incoming data is determined not to be a packed executable.

14. (Currently Amended) The system of Claim 1, wherein the incoming data can be intercepted from at least one data source including a computer network~~, and~~ or distributable media further including a floppy disk, a flash memory storage device, a CD-ROM disk, a magnetic tape, or combinations thereof.

15. (Previously Presented) The system of Claim 1, further comprising, first determining whether the incoming data is known malware before determining if the incoming data is a packed executable, and if not, then determining if the incoming data is a packed executable.

16. (Previously Presented) The system of Claim 15, wherein anti-virus software can be employed in determining whether the incoming data is malware.

17. (Previously Presented) The system of Claim 16, wherein the determining by anti-virus software can be by signature or pattern recognition processes.

18. (Previously Presented) An electronic device comprising the system of Claim 1, such that the electronic device can be placed between a network and a computer device to facilitate intercepting data directed to a computing device.

19. (Previously Presented) The method of Claim 4, further comprising, first determining whether the incoming data is known malware before determining if the incoming data is a packed executable.

20 (Currently amended) The method of Claim 4, wherein generating ~~an~~ a corresponding substitute unpacked executable at least in part employs ~~an~~ a substitute unpacker code segment other than the ~~loader/~~unpacker received with the first packed executable.

21. (Currently amended) The method of Claim 20, wherein the employed substitute unpacker code is selected from a group of at least one modularized substitute unpacker modules germane to unpacking a first packed executable of a particular type and further germane to unpacking a first packed executable that has been intercepted.

22 (Previously Presented) The method of Claim 4, wherein intercepting incoming data intercepts data as it arrives at the computing device from a network or a distributable media.

23. (Currently amended) The method of Claim 4, wherein generating the corresponding substitute unpacked executable occurs without executing any portion of the unpacked executable.

24. (Currently amended) The method of Claim 4, wherein the corresponding unpacked executable corresponds to a complete first packed executable and not just a portion thereof.